Enhancing Language Model with Both Human and Artificial Intelligence Feedback Data

Haoting Zhang, Jinghai He, Jingxu Xu, Jingshen Wang, Zeyu Zheng
University of California, Berkeley

For enterprises that face data privacy regulation or other constraints, instead of just using the closed-source pre-trained language models, training or fine-tuning a language model of their own have observed increasing needs. For a number of applications, training a language model would need excessive amount of data based on human feedback, which sometimes can be too expensive or not adequately accessible. In this work, we propose a simulation optimization framework to train the language model, using not only the human feedback data but also feedback data provided by pre-trained artificial intelligence (AI) models. When integrating AI data to human data to enhance model training, we employ the method of control variate for variance reduction. Because AI data and human data have various accessing costs and data quality, we provide a procedure to evaluate how to allocate a budget to assign to different data sources, in order to maximize the overall training performance. Numerical experiments demonstrate that our proposed procedure enhances the performance of the language model.

1. INTRODUCTION

In recent years, language models, such as the GPT (Generative Pre-trained Transformer) series from OpenAI (2023), have been widely used in a variety of applications, ranging from enhancing customer service through chatbots to supporting complex decision-making processes in business and healthcare. Instead of solely relying on pre-trained language models, some enterprises, facing data privacy regulations or other constraints, opt to train or fine-tune open-source pre-trained language models specifically for their own use (De Andrade and Tumelero 2022, Skiles 2023). Fine-tuning language models with their own data allows these enterprises to tailor the models to meet their unique operational needs and industry-specific challenges. During the fine-tuning process, a small portion of the model's parameters are adapted, enhancing the performance of the language model on tailored tasks (Radiya-Dixit and Wang 2020). Regarding the enhancement of these language models, human feedback is collected for the pre-trained models to learn and adapt. This strategy enables the trained models to align with human values and preferences (Christiano et al. 2017). On the other hand, although learning from human feedback has achieved success in practice, it also presents challenges. First, instructing humans to provide feedback is time-consuming and resource-intensive, which sometimes is not affordable for small businesses or non-profit organizations. Second, learning from human feedback generally depends on a small pool of humans

and their subjective preferences, raising concerns about fairness and inclusiveness. Lastly, as the demand for language model services grows, the necessity to rapidly train models for varied tasks becomes more pressing, while relying on human feedback can limit the scalability and adaptability of training language models.

To address the challenges brought by learning from human feedback, a feasible framework is to learn from artificial intelligence (AI) feedback (Bai et al. 2022b). In this framework, the language model learns from feedback generated by other AI models. These AI models, trained to imitate human evaluative patterns and preferences, have proven to align with human values and preferences to a significant extent. This AI-driven approach not only mitigates the cost and resource constraints associated with human feedback but also offers a scalable and more objective method for improving language models. However, learning from AI feedback is not without its own challenges. Firstly, AI feedback may lack the depth of empathy inherent to human responses, potentially leading to models that are less nuanced in handling complex emotional contexts. Also, the effectiveness of learning from AI feedback is largely constrained by the limitations of the AI models themselves. The limitations of relying solely on either human or AI feedback underscore the importance of integrating both sources to optimally enhance language models.

1.1. Method and Results

In this work, we propose a simulation optimization framework to enhance language models with both human and AI preference data. Specifically, the objective function is to maximize the mean likelihood function of the language model generating outputs that align with human preferences. The decision variable to optimize is the set of parameters in the language model. We regard acquiring preference data from humans/AI as simulating a sample from a stochastic system. Moreover, we consider human preference data as "high fidelity", whereas data collected from AI models is treated as "low fidelity". To approximate the mean likelihood (our objective function), we benefit from variance reduction and employ the method of *control variate* (Asmussen and Glynn 2007). Specifically, we use the human preference data to construct the sample mean of the objective function. We then use the AI preference data to reduce the variance of the sample mean. An illustration of our framework is summarized in **Figure 1**. In addition, we consider multiple AI models in our work and sort them in descending order based on "fidelity" (Zheng and Glynn 2017, Zheng et al. 2018). We approximate the objective function recursively, using lower-fidelity AI models to reduce the variance of higher-fidelity AI models. Futhermore, given that AI data and human data have different access costs and quality levels, we provide a procedure to evaluate how to allocate a budget across different data sources to maximize overall training performance. Specifically, we aim to minimize the variance of the constructed training objective function, and facilitate the budget allocation procedure by solving a nonlinear integer programming problem.

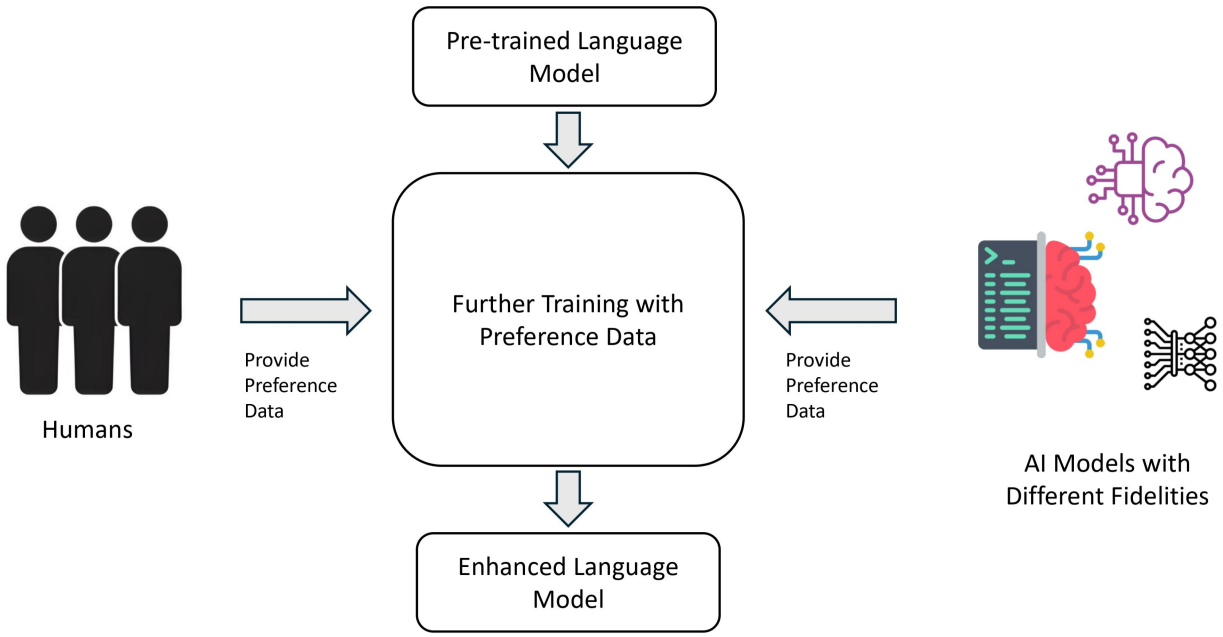

Figure 1 **An illustration of the framework of enhancing a language model using both human and AI preference data.**

Our contribution is summarized as follows:

1. We propose a framework to enhance language models using both human and AI preference data. This framework employs the method of control variate and constructs an objective function that is unbiased and has a lower variance. We also prove the consistency of the learning procedure associated with our proposed framework.

2. We conduct numerical experiments to demonstrate the efficacy of our proposed framework. Specifically, we show through experiments that our framework outperforms methods relying solely on either human or AI feedback. Furthermore, the experimental results suggest that involving more AI models to provide feedback also enhances the performance of language models.

3. Although our focus in this work is on enhancing language models, our proposed framework is applicable to other simulation optimization problems where samples of different fidelities can be acquired. For example, in applications of queuing systems and financial systems, the objective function may involve solutions of stochastic differential equations, where samples of different fidelities can be constructed through time discretization at different resolutions (Xu et al. 2014, Xu and Zheng 2023).

1.2. Literature Review

Training language models through learning from human feedback has become a widely adopted methodology. This approach ensures that the models are better aligned with human preferences

and can generate responses that are more contextually relevant to users' requirements. For example, Christiano et al. (2017) introduce a framework for training language models, named *reinforcement learning from human feedback* (RLHF). In this framework, a reward model is first learned using human feedback, and then the language model is trained with this learned reward model. Rafailov et al. (2024) simplify the RLHF framework and propose learning the language model directly through the provided human preference data. On the other hand, learning from human feedback requires extensive time and resources, and has the risk of exposing humans to harmful content. To overcome these shortcomings, Bai et al. (2022b) propose utilizing feedback data from AI models. Specifically, selected AI models are employed to substitute humans in providing preferences between contents.

Our proposed learning framework is connected to variance reduction methods in simulation. Variance reduction, aimed at decreasing the variability of estimators constructed by simulated samples, enhances the efficiency of approximation for unknown quantities. In the context of variance reduction, prominent methods include but are not limited to importance sampling (Liu 2015, Tong and Liu 2016, Feng and Song 2019, He et al. 2023, Bai et al. 2023, Deo and Murthy 2023), control variate (Kim and Henderson 2007, Peherstorfer et al. 2016), and stratification (Rhee and Glynn 2015, Vihola 2018).

Our work also benefits from simulation optimization. The strategies for solving the simulation optimization problems depend largely on the features of the objective function and the feasible set. If the feasible set is discrete, the methodologies utilized can be found in the broad literature of discrete optimization via simulation; see Luo et al. (2015), Fan et al. (2020), Hong et al. (2022) among others. When the feasible set is continuous, under different circumstances, various methods are developed, including but are not limited to gradient-based methodologies (Ahamed et al. 2006, Zhu and Dong 2021, Peng et al. 2022) and meta-model based methods (Dong et al. 2018, L. Salemi et al. 2019, Xie et al. 2020, Semelhago et al. 2021, Hong and Zhang 2021, Wang et al. 2023).

2. PROBLEM STATEMENT

In this section, we formalize the problem of enhancing a language model using both human and artificial intelligence (AI) feedback. We also provide the preliminaries of our method and set up the notation. We aim to enhance the performance of a pre-trained language model using the preference data. The data are collected from both humans and other AI models. The language model is represented by a policy

$$\pi_\theta(y \mid x).$$

Here x denotes the user input to the language model (also known as the *prompt*), y is the output generated by the language model, and $\theta \in \Theta$ is the parameters of the pre-trained language model.

The policy $\pi_\theta(y \mid x)$ defines a probability of generating the output y conditional on the input x, with a fixed value of parameters θ. In this work, our goal is not to train a language model from scratch. Instead, we focus on enhancing (also known as *fine-tuning*) a pre-trained language model using preference data. This means that we will not alter the model's structure (e.g., the fixed structure of neural networks) that represents $\pi_\theta(y \mid x)$, but will instead adjust its parameters $\theta \in \Theta$. We denote the current parameter of the pre-trained language model as $\theta^{(0)}$. Utilizing the feedback data, we then further optimize the parameters θ to better align the language model with human preferences.

Data Set Generation

We consider the scenario when the data set of contexts for comparison is generated by a language model. The language model for data generation can be either 1) the pre-trained language model we would like to enhance or 2) another different language model. Specifically, we denote

$$\boldsymbol{z}^{(i)} \doteq \left(x^{(i)}, y_1^{(i)}, y_2^{(i)}\right) \overset{i.i.d.}{\sim} \mathcal{D}. \tag{1}$$

Here $\boldsymbol{z}^{(i)}$ represents a data point generated by the language model and is independent and identically distributed (i.i.d.) from the generation distribution $\mathcal{D}$. Furthermore, in each data point, $x^{(i)}$ denotes the "prompt" that instructs the language model to generate outputs. Also, $y_1^{(i)}$ and $y_2^{(i)}$ represent two generated contexts under the instruction of $x^{(i)}$. These two generated contexts are further compared by human and/or AI models. We let Ω denote the support of the distribution $\mathcal{D}$.

Objective

We here describe the training objective of language models using the data collected as (1). Specifically, the language model is trained to align with human preference. Thus, for each $\boldsymbol{z}^{(i)}$, human is involved to provide the preference between $y_1^{(i)}$ and $y_2^{(i)}$. Then, the language model $\pi_\theta(y \mid x)$ is further trained to generate the context that are more preferred with high probabilities. Without loss of generality, we assume $y_1^{(1)}$ is always preferred to $y_2^{(1)}$ for humans. That is, $y_1^{(i)} \succ y_2^{(i)} \, \forall i$.

With the human preference data $D_0 = \left\{\left(x^{(1)}, y_1^{(1)}, y_2^{(1)}\right), \left(x^{(2)}, y_1^{(2)}, y_2^{(2)}\right), \ldots, \left(x^{(N_0)}, y_1^{(N_0)}, y_2^{(N_0)}\right)\right\}$, the language model is further trained by

$$\theta^* = \arg\min_{\theta \in \Theta} \left\{ L(\theta) \doteq \mathbb{E}_{\mathcal{D}}\left[f\left(\boldsymbol{z}^{(i)}, \theta\right)\right]\right\}. \tag{2}$$

Here $\theta \in \Theta$ represents the parameters in the language model to be optimized, $f\left(\boldsymbol{z}^{(i)}, \theta\right)$ represents the loss function of each data point $\boldsymbol{z}^{(i)}$ with explicit preference, and the distribution $\mathcal{D}$ is approximated by the data set D_0. In this work, we specifically select the loss function

$$f\left(\boldsymbol{z}^{(i)}, \theta\right) = -\log\sigma\left(\beta \log \frac{\pi_\theta\left(y_1^{(i)} \mid x\right)}{\pi_{\theta^{(0)}}\left(y_1^{(i)} \mid x\right)} - \beta \log \frac{\pi_\theta\left(y_2^{(i)} \mid x\right)}{\pi_{\theta^{(0)}}\left(y_2^{(i)} \mid x\right)}\right).$$

Here $\sigma(r) = \frac{1}{1+e^{-r}}$ is the sigmoid function, β is a pre-selected hyperparameter, $\pi_\theta(y \mid x)$ denotes the language model we aim to enhance. Also, $\pi_{\theta^{(0)}}(y \mid x)$ is the pre-trained language model, serving as the baseline for enhancing the language model. This loss function indicates the negative likelihood function associated with the Bradley-Terry model (Hunter 2004). This model captures the human preferences as

$$\mathbb{P}^*\left(y_1^{(i)} \succ y_2^{(i)} \mid x\right) = \left(1 + \exp\left(\beta \log \frac{\pi^*\left(y_2^{(i)} \mid x\right)}{\pi_{\theta^{(0)}}\left(y_2^{(i)} \mid x\right)} - \beta \log \frac{\pi^*\left(y_1^{(i)} \mid x\right)}{\pi_{\theta^{(0)}}\left(y_1^{(i)} \mid x\right)}\right)\right)^{-1}.$$

Here $\mathbb{P}^*\left(y_1^{(i)} \succ y_2^{(i)} \mid x\right)$ is the ground-truth probability that humans prefer $y_1^{(i)}$ over $y_2^{(i)}$, and $\pi^*(y \mid x)$ denotes the language model that exactly aligns with human preferences. For more details on this loss function and other loss functions used to enhance a language model, please refer to Rafailov et al. (2024).

Besides the human preference data, our work also considers preference data provided by AI models. Specifically, instead of focusing on a single AI model, we consider a series of K AI models, denoted by $\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_K$. Furthermore, we assume these AI models are sorted in descending order based on 'fidelity'. That is, $\boldsymbol{G}_k$ exhibits more similarity with human preferences than $\boldsymbol{G}_{k'}$ when $k < k'$. We note that quantifying the similarity between an AI model $\boldsymbol{G}_k$ and human preferences is generally challenging. However, there are some ranking lists of different AI models that we can refer to. Additionally, the cost of applying AI models with higher fidelity is generally higher than those with lower fidelity. Given $\boldsymbol{z}^{(i)}$ as in (1), the AI model $\boldsymbol{G}_k$ provides the preference $y_{(k);1}^{(i)} \succ y_{(k);2}^{(i)}$. To indicate the difference in preferences between an AI model $\boldsymbol{G}_k$ and humans, we denote a set

$$S_k = \left\{\boldsymbol{z}^{(i)} \in \Omega \mid y_{(k);1}^{(i)} = y_2^{(i)}, y_{(k);2}^{(i)} = y_1^{(i)}\right\}.$$

That is, S_k includes the data points for which the AI model and humans have opposite preferences. The loss function based on the AI's preference is then formulated by

$$\begin{aligned}\tilde{f}_k\left(\boldsymbol{z}^{(i)} = \left(x^{(i)}, y_1^{(i)}, y_2^{(i)}\right), \theta\right) =& \mathbb{I}\left\{\boldsymbol{z}^{(i)} \in S_k\right\} f\left(\left(x^{(i)}, y_2^{(i)}, y_1^{(i)}\right), \theta\right) + \left(1 - \mathbb{I}\left\{\boldsymbol{z}^{(i)} \in S_k\right\}\right) f\left(\boldsymbol{z}^{(i)}, \theta\right) \\ =& f\left(\left(x^{(i)}, y_{(k);1}^{(i)}, y_{(k);2}^{(i)}\right), \theta\right).\end{aligned} \tag{3}$$

In this manner, when the AI model has the same preference as humans, we have $\tilde{f}_k = f$. If the AI model aligns with human preferences in most scenarios, the objective function associated with AI preference, $\tilde{f}_k$, then serves as an approximation for that of humans, f. In this work, we assume that

$$\operatorname{Cov}\left[f\left(\boldsymbol{z}^{(i)}, \theta\right), \tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta\right)\right] > 0 \quad \forall \theta \in \Theta, \forall k \in \{1, 2, \ldots, K\}.$$

Also, although the set S_k is generally intractable and unknown, the loss function associated with an AI model can still be constructed based on the AI's preference as in (3).

3. METHODOLOGY

In this section, we provide the procedure for enhancing a language model using the preference data from both human and artificial intelligence (AI) models. Specifically, we construct a sequence of sets

$$D_0 \subseteq D_1 \subseteq D_2 \subseteq \ldots \subseteq D_K,$$

where each D_k contains $\boldsymbol{z}^{(i)}$'s drawn from the distribution $\mathcal{D}$, and we denote by $N_k \doteq |D_k|$ the number of data points in each set. Moreover, D_0 is for humans to provide preferences, and D_k is for the AI model $\boldsymbol{G}_k$ with $k \in \{1, 2, \ldots, K\}$. In Section 3.1, we construct the objective function, incorporating the preference data in hand, to enhance the language model and provide the procedure for optimizing it. In Section 3.2, we describe the experimental design, which includes deciding 1) the sample size for each D_k and 2) the hyperparameters in the constructed objective function.

3.1. Objective Function & Optimization

In this section, we describe the procedure of enhancing the language model with the preference data in hand. We postpone the acquisition of preference data in Section 3.2. By integrating feedback from both human and AI models, we construct the objective function to minimize as

$$\tilde{L}(\theta) = \frac{1}{N_0} \sum_{\boldsymbol{z}^{(i)} \in D_0} f\left(\boldsymbol{z}^{(i)}, \theta\right) + \sum_{k=1}^{K} \alpha_k \left(\frac{1}{N_k} \sum_{\boldsymbol{z}^{(i)} \in D_k} \tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta\right) - \frac{1}{N_{k-1}} \sum_{\boldsymbol{z}^{(i)} \in D_{k-1}} \tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta\right) \right), \quad (4)$$

where $f\left(\boldsymbol{z}^{(i)}, \theta\right)$ is the loss function associated with human preference and $\tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta\right)$ is the loss function for the AI model $\boldsymbol{G}_k$'s preference. Furthermore, $\alpha_k > 0$'s are hyperparameters that are pre-selected, and we postpone the discussion to Section 3.2.

The objective function (4) takes advantage of *control variate*, which is a technology for variance reduction using simulated samples to approximate an expectation; see Asmussen and Glynn (2007) and Peherstorfer et al. (2016). That is, we employ the correlated samples $\tilde{f}_1\left(\boldsymbol{z}^{(i)}, \theta\right)$ of the variance of the empirical loss $\frac{1}{N_0} \sum_{\boldsymbol{z}^{(i)} \in D_0} f\left(\boldsymbol{z}^{(i)}, \theta\right)$ when approximating $\mathbb{E}_{\boldsymbol{z}^{(i)} \sim \mathcal{D}}\left[f\left(\boldsymbol{z}^{(i)}, \theta\right)\right]$. Furthermore, since the mean value $\mathbb{E}_{\boldsymbol{z}^{(i)} \sim \mathcal{D}}\left[\tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta\right)\right]$ is unknown and requires approximation by $\frac{1}{N_k} \sum_{\boldsymbol{z}^{(i)} \in D_k} \tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta\right)$, we then use $\tilde{f}_{k+1}\left(\boldsymbol{z}^{(i)}, \theta\right)$ to reduce the associated variance recursively.

PROPOSITION 1. *Regarding the objective function (4), we have*

$$\mathbb{E}\left[\tilde{L}(\theta)\right] = \mathbb{E}_{\boldsymbol{z}^{(i)} \sim \mathcal{D}}\left[f\left(\boldsymbol{z}^{(i)}, \theta\right)\right] \quad \forall \theta \in \Theta$$

and

$$\mathrm{Var}\left[\tilde{L}(\theta)\right] < \mathrm{Var}\left[\frac{1}{N_0} \sum_{\boldsymbol{z}^{(i)} \in D_0} f\left(\boldsymbol{z}^{(i)}, \theta\right)\right] \quad \forall \theta \in \Theta.$$

That is, the objective function (4) is an unbiased estimator of the mean loss function, and reduces the variance of the empirical loss associated with the human preference data.

As documented by existing literature, reducing variance during the learning process of machine learning models offers advantages. Specifically, Johnson and Zhang (2013) propose the algorithm stochastic variance reduced gradient to accelerate the convergence rate of the learned model. Also, a trend of research focuses on regularization technologies to address the bias-variance trade-off of the learned model (Hastie et al. 2009). This trade-off reduces the risk of overfitting, ensuring better generalization to unseen data. In this work, we treat AI preferences as correlated samples of human preferences. To this end, we employ the control variate method to reduce the variance of the empirical loss function—our objective function for training the language model. We construct this objective function to minimize variance. The detailed procedure is postponed to Section 3.2.

The objective function (4) involves the language model $\pi_\theta(y \mid x)$, which is represented by neural networks with complex structures. Thus, minimizing such an objective function is generally challenging and does not yield an explicit solution. In our work, we specifically choose the stochastic gradient descent method to facilitate the optimization process. In terms of approximating the gradient of the objective function, we utilize the backpropagation algorithm; see Goodfellow et al. (2016) for a detailed overview.

We now establish the consistency of our proposed learning procedure. For ease of notation, we consider the scenario where $K = 1$, meaning there is one AI model used to provide preference data. Our theoretical results can be generalized to multiple AI models without essential difficulty. We assume the following conditions:

ASSUMPTION 1.

1. *The feasibility set* Θ *is compact.*
2. *There exist function* $\mathcal{L} : \Omega \mapsto \mathbb{R}^+$ *such that for almost every* $\boldsymbol{z}^{(i)}$ *and all* $\theta_1, \theta_2 \in \Theta$,

$$\left| f(\boldsymbol{z}^{(i)}, \theta_1) - f(\boldsymbol{z}^{(i)}, \theta_2) \right| \leqslant \mathcal{L}(\boldsymbol{z}^{(i)}) \left\| \theta_1 - \theta_2 \right\|$$

and

$$\left| \tilde{f}_1(\boldsymbol{z}^{(i)}, \theta_1) - \tilde{f}_1(\boldsymbol{z}^{(i)}, \theta_2) \right| \leqslant \mathcal{L}(\boldsymbol{z}^{(i)}) \left\| \theta_1 - \theta_2 \right\|.$$

The function $\mathcal{L}$ *satisfies* $\mathbb{E}_{\boldsymbol{z}^{(i)} \sim \mathcal{D}} \left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right) \right] < \infty$.

THEOREM 1 **(consistency)**. *Denote* $\tilde{L}^*_{N_0,N_1} = \min_{\theta \in \Theta} \tilde{L}(\theta)$, *and* $L^* = \min_{\theta \in \Theta} \mathbb{E}_{\mathcal{D}} \left[f\left(\boldsymbol{z}^{(i)}, \theta\right) \right]$. $\hat{\theta}_{N_0,N_1} = \arg\min_{\theta \in \Theta} \tilde{L}(\theta)$ *represents the point at which* $\tilde{L}(\theta)$ *is minimized. Under Assumption 1, we have*

$$\lim_{N_0 \to +\infty} \tilde{L}^*_{N_0,N_1} = L^* \quad w.p.1.$$

and

$$\lim_{N_0 \to +\infty} \hat{\theta}_{N_0,N_1} = \theta^* \quad w.p.1.,$$

where "w.p.1." stands for "with probability one".

D enote $B(\theta,\delta)$ as the open ball with center θ and radius δ. Given any $\epsilon>0$, since Θ is compact, we can choose a finite collection of points $\{\theta_1,\theta_2,\ldots,\theta_r\}$ such that $\Theta\subset\cup_{j=1}^{r}B\left(\theta_j,\frac{\epsilon}{2(1+2\alpha_1)\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]}\right)$. For convenience denote $B_j=B\left(\theta_j,\frac{\epsilon}{2(1+2\alpha_1)\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]}\right)$. By Lipschitz continuity assumption, for every $j=1,2,\ldots,r$,

$$\sup_{\theta\in\Theta\cap B_j}\left|\tilde{L}(\theta)-\tilde{L}\left(\theta_j\right)\right|\leqslant\left(\frac{1+\alpha_1}{N_0}\sum_{\boldsymbol{z}^{(i)}\in D_0}\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)+\frac{\alpha_1}{N_1}\sum_{\boldsymbol{z}^{(i)}\in D_1}\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right)\frac{\epsilon}{2\left(1+2\alpha_1\right)\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]}.$$

By strong law of large numbers (SLLN), $\frac{1}{N_0}\sum_{\boldsymbol{z}^{(i)}\in D_0}\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)$ converges to $\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]$ a.s. as $N_0\to+\infty$. Since $N_1\leqslant N_0$, $\frac{1}{N_1}\sum_{\boldsymbol{z}^{(i)}\in D_1}\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)$ also converges to $\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]$ a.s. as $N_0\to+\infty$. Therefore, for sufficiently large N_0, we have

$$\sup_{\theta\in\Theta\cap B_j}\left|\tilde{L}(\theta)-\tilde{L}\left(\theta_j\right)\right|\leqslant\left(2\left(1+\alpha_1\right)\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]+2\alpha_1\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]\right)\frac{\epsilon}{2\left(1+2\alpha_1\right)\mathbb{E}\left[\mathcal{L}\left(\boldsymbol{z}^{(i)}\right)\right]}=\epsilon,\quad j=1,2,\ldots,r$$

w.p.1. According to strong law of large number, for every $\theta\in\Theta$,

$$\lim_{N_0\to+\infty}\tilde{L}(\theta)=\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta\right)\right]+\alpha_1\left(\mathbb{E}\left[\tilde{f}_1\left(\boldsymbol{z}^{(i)},\theta\right)\right]-\mathbb{E}\left[\tilde{f}_1\left(\boldsymbol{z}^{(i)},\theta\right)\right]\right)=\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta\right)\right],\text{w.p.1.}$$

Because r is finite, for given $\epsilon>0$, there exists sufficiently large N_0 such that

$$\sup_{j=1,2,\ldots,r}\left|\tilde{L}(\theta_j)-\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta_j\right)\right]\right|\leqslant\epsilon,\text{w.p.1.}$$

Consider now an arbitrary point $\theta\in\Theta$. By the construction of B_j, there exists some $\theta_j\in\Theta$ and is the center of B_j, such that $\theta\in B_j$. Therefore for sufficiently large N_0 independent of θ, we have

$$\begin{aligned}\left|\tilde{L}(\theta)-\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta\right)\right]\right|&\leqslant\left|\tilde{L}(\theta)-\tilde{L}(\theta_j)\right|+\left|\tilde{L}(\theta_j)-\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta_j\right)\right]\right|+\left|\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta\right)\right]-\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta_j\right)\right]\right|\\&\leqslant\epsilon+\epsilon+\mathbb{E}\left[\left|f(\boldsymbol{z}^{(i)},\theta)-f(\boldsymbol{z}^{(i)},\theta_j)\right|\right]\leqslant 3\epsilon.\end{aligned}$$

So the uniform convergence is proved, i.e. $\sup_{\theta\in\Theta}\left|\tilde{L}(\theta)-\mathbb{E}\left[f\left(\boldsymbol{z}^{(i)},\theta\right)\right]\right|\to 0$ a.s. when $N_0\to+\infty$. The consistency of $\tilde{L}^*_{N_0,N_1}$ and $\hat{\theta}_{N_0,N_1}$ can be then proved based on Theorem 5.3 in Shapiro et al. (2021).

3.2. Experimental Design

In this section, we describe the experimental design, including 1) deciding the sample size of each preference dataset, $\{N_k\}_{k=0}^{K}$ and selecting the hyperparameters in the objective function (4), $\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\ldots,\alpha_K)$. Here we aim to minimize the mean squared error (MSE) of the loss function $\tilde{L}(\theta)$ at the optimal parameters θ^*. Since $\tilde{L}(\theta)$ is an unbiased estimation of $L(\theta)$, MSE is exactly the variance of $L(\theta)$. Specifically, we have

$$\text{MSE}\left(\tilde{L}(\theta)\right)\doteq\mathbb{E}\left[\tilde{L}(\theta^*)-L(\theta^*)\right]^2=\frac{\sigma_0^2}{N_0}+\sum_{k=1}^{K}\left(\frac{1}{N_{k-1}}-\frac{1}{N_k}\right)\left(\sigma_k^2\alpha_k^2-2C_k\alpha_k\right),\tag{5}$$

where $\sigma_0^2 \doteq \mathrm{Var}\left[f\left(\boldsymbol{z}^{(i)}, \theta^*\right)\right], \sigma_k^2 \doteq \mathrm{Var}\left[\tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta^*\right)\right]$ $k \geqslant 1$, and $C_k \doteq \mathrm{Cov}\left[f\left(\boldsymbol{z}^{(i)}, \theta^*\right), \tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta^*\right)\right]$. In practice, these statistical quantities are unknown and require to be estimated from the data. Therefore, regarding the experimental design, we first conduct a warm-up procedure:

1. Randomly select $D^{(0)} \doteq \left\{\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}, \ldots, \boldsymbol{z}^{(m_0)}\right\}$ and acquire preference from human and each AI model regarding $\forall \boldsymbol{z}^{(i)} \in D^{(0)}$;

2. Update the language model using the human preference data by

$$\theta^{(1)} = \arg\min_{\theta \in \Theta} \frac{1}{m_0} \sum_{i=1}^{m_0} f\left(\boldsymbol{z}^{(i)}, \theta\right);$$

3. Construct the loss functions

$$\left\{f\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right), \tilde{f}_1\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right), \ldots, \tilde{f}_K\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right)\right\}_{i=1}^{m_0};$$

4. Estimate the quantities as $\widehat{\sigma_0^2} = \frac{1}{m_0 - 1} \sum_{i=1}^{m_0} \left(f\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right) - \frac{1}{m_0} \sum_{j=1}^{m_0} f\left(\boldsymbol{z}^{(j)}, \theta^{(1)}\right)\right)^2$,

$$\widehat{\sigma_k^2} = \frac{1}{m_0 - 1} \sum_{i=1}^{m_0} \left(\tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right) - \frac{1}{m_0} \sum_{j=1}^{m_0} \tilde{f}_k\left(\boldsymbol{z}^{(j)}, \theta^{(1)}\right)\right)^2,$$

and

$$\widehat{C_k} = \frac{\sum_{i=1}^{m_0} \left(\left(f\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right) - \frac{1}{m_0} \sum_{j=1}^{m_0} f\left(\boldsymbol{z}^{(j)}, \theta^{(1)}\right)\right)\left(\tilde{f}_k\left(\boldsymbol{z}^{(i)}, \theta^{(1)}\right) - \frac{1}{m_0} \sum_{j=1}^{m_0} \tilde{f}_k\left(\boldsymbol{z}^{(j)}, \theta^{(1)}\right)\right)\right)}{(m_0 - 1)}$$

for any $k \in \{1, 2, \ldots, K\}$.

Furthermore, either instructing humans or invoking AI models to provide a preference brings cost. We consider the cost when minimizing $\mathrm{MSE}\left(\tilde{L}(\theta)\right)$ with a given budget of W. Regarding the acquisition of a preference data point, we denote the cost associated with the AI model $\boldsymbol{G}_k$ by w_k and the cost associated with humans by w_0. To begin with, we first consider a scenario when some open-source AI models can provide preference data without any cost. Specifically, we assume that $\boldsymbol{G}_{k_f}$ is such an AI model with $k_f = \min\{k \mid w_k = 0\}$. In this scenario, we let N_{k_f} sufficiently large if the computational cost is not a concern. We then have an accurate approximation for $\mathbb{E}\left[\tilde{f}_{k_f}\left(\boldsymbol{z}^{(i)}\right)\right]$. Recall that, in the objective function (4), the preference data from AI model $\boldsymbol{G}_{k_f+1}$ are used to reduce the variance of the empirical loss $\frac{1}{N_{k_f}} \sum_{i=1}^{N_{k_f}} \tilde{f}_{k_f}\left(\boldsymbol{z}^{(i)}\right)$. Since now the variance approaches 0, there is no need to acquire preference data from $\boldsymbol{G}_{k_f+1}$, as well as any other AI model $\boldsymbol{G}_{k'}$ with $k' > k_f$. Therefore, when deciding the sample sizes of preference data, $\left\{N_0, N_1, \ldots, N_{k_f-1}\right\}$ are taken into consideration. Without loss of generality, we assume that $w_k > 0$ in the following discussion.

Given the cost of acquiring preference data from humans and each AI model $\{w_k\}_{k=0}^K$, as well as the total budget W, the sample size $\{N_k\}_{k=0}^K$ and the hyperparameters $\boldsymbol{\alpha}$ are determined by

solving the following optimization problem. This problem incorporates the estimated quantities $\left\{\widehat{\sigma_k^2}\right\}_{k=0}^K$ and $\left\{\widehat{C}_k\right\}_{k=1}^K$, as substituted into (5):

$$\begin{aligned} \underset{\boldsymbol{\alpha}\in\mathbb{R}_+^K; N_0,N_1,\ldots,N_K\in\mathbb{N}}{\text{minimize}} \quad & \frac{\widehat{\sigma_0^2}}{N_0}+\sum_{k=1}^K\left(\frac{1}{N_{k-1}}-\frac{1}{N_k}\right)\left(\widehat{\sigma_k^2}\alpha_k^2-2\widehat{C_k}\alpha_k\right) \\ \text{subject to} \quad & N_k\geqslant m_0,\ k=0,1,\ldots,K, \\ & N_{k-1}\leqslant N_k,\ k=1,2,\ldots,K, \\ & \sum_{k=0}^K w_k N_k\leqslant W. \end{aligned} \tag{6}$$

The optimization problem (6) is a nonlinear mixed integer programming. In general, there are no closed-form solutions. On the other hand, the optimal solution regarding $\boldsymbol{\alpha}$ does not depend on the selection of $N_0, N_1, \ldots, N_K$. Thus, we first attain $\boldsymbol{\alpha}^* = (\alpha_1^*, \alpha_2^*, \ldots, \alpha_K^*)$ with $\alpha_k^* = \widehat{C_k}/\widehat{\sigma_k^2}$. We then plug $\boldsymbol{\alpha}^*$ in the optimization problem (6) so that it reduces to a nonlinear integer programming. Nonlinear integer programming problems can be generally solved by the branch and bound approach or heuristic methods (e.g., simulated annealing). For detailed procedures of solving nonlinear integer programming problems, we refer to Li and Sun (2006). With a slight abuse of notation, we denote the optimal solution of (6) by $\{N_k\}_{k=0}^K$ in the remainder of the text. After deciding $\{N_k\}_{k=0}^K$, we acquire the preference data from humans and $\boldsymbol{G}_k$'s, and then construct the objective function (4) with $\boldsymbol{\alpha}^*$ plug-in. After optimizing the objective function (4) as in Section 3.1, we facilitate enhancing the language model using the preference data from both humans and AI models.

4. EXPERIMENTS

In this section, we conduct numerical experiments to perform the proposed procedure for enhancing language models. The experimental settings are summarized as follows:

1. Regarding the initial pre-trained language model, we select TinyLlama (Zhang et al. 2024). In terms of the artificial intelligence (AI) models, we select ChatGPT 4, ChatGPT 3.5 Turbo (OpenAI 2023), and Llama2 (Touvron et al. 2023).

2. We utilize an open-source preference dataset for training the language model, where the preference has been decided by humans (Bai et al. 2022a). For AI preferences, we input the pair of contexts to the AI models for comparison. The sample size for the human preference data D_0 is fixed to be 1000. In addition, we have $D_1 = 1500, D_2 = 2000, D_3 = 2500$.

3. The compared procedures of training language models include 1) our procedure with $K = 1$ AI model, 2) our procedure with $K = 3$ AI models, 3) the procedure that entirely relies on human preference data, 4) the procedure that entirely relies on the preference data provided by the highest-fidelity AI model, and 5) the procedure with the initial language model without further training.

Framework	Discrete Agreement Mean Value	Discrete Agreement Standard Deviation
Our Procedure ($K=1$)	30.72	2.39
Our Procedure ($K=3$)	31.94	2.87
Procedure with Human Preference	29.36	2.57
Procedure with AI Preference (Highest Fidelity)	27.91	1.98
Initial Language Model without Further Training	27.15	1.32

Table 1 Experimental results of the language models' performance with different training procedures.

4. To evaluate the performance of the language model, we consider a metric named *discrete agreement* introduced in Nie et al. (2024), which is the accuracy of the language model's judgment towards the human-labeled dataset (Nie et al. 2024). Specifically, the dataset contains 80 pairs of questions and answers, with each answer being either "yes" or "no". Each question is input into the AI model, which then answers "yes" or "no". The value of discrete agreement is the ratio of answers provided by the AI that are consistent with those in the dataset. A higher value of discrete agreement indicates a better performance of the language model.

5. Our experiments were conducted with Pytorch and Python 3.8 on a computer equipped with two AMD Ryzen Threadripper 3970X 32-Core Processors, 256 GB memory, and two Nvidia GeForce RTX 3090 GPUs with 24GB of RAM each.

The numerical results are contained in **Table 1**. The recorded mean values and standard deviation are based on running the experiment 5 times. The experimental results provide the following insights: First, compared to the initial language model without further training, incorporating preference data from either humans or AI models enhances the performance of the language model. Second, incorporating both human and AI feedback outperforms methods that rely entirely on feedback from either source alone. Lastly, incorporating feedback from additional AI models also enhances the performance of the language model.

5. CONCLUSION

In this work, we consider enhancing language models using both human and artificial intelligence (AI) preference data. We propose a simulation optimization framework where samples (preference data) are acquired with different fidelities to reduce the variance of the approximated objective function. We conclude our work by outlining potential future work. First, our procedure determines the sample size for each dataset by minimizing the variance of the objective function, a process that involves quantities requiring approximation with samples acquired during the warm-up stage.

It remains a question how to allocate the total number of samples in the warm-up stage to accurately approximate these quantities while reserving a sufficient budget for subsequent sample size allocation. Furthermore, our framework reduces the variance of the objective function when incorporating AI preference data alongside human preference data. Alternative methods for constructing objective functions to train language models might also prove effective.

References

Ahamed, T. I., Borkar, V. S., and Juneja, S. (2006). Adaptive importance sampling technique for Markov chains using stochastic approximation. Operations Research, 54(3):489–504.

Asmussen, S. and Glynn, P. W. (2007). Stochastic simulation: algorithms and analysis, volume 57. Springer.

Bai, Y., Huang, Z., Lam, H., and Zhao, D. (2023). Overconservativeness of variance-based efficiency criteria and probabilistic efficiency in rare-event simulation. Management Science.

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al. (2022a). Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv preprint arXiv:2204.05862.

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. (2022b). Constitutional AI: Harmlessness from AI feedback. arXiv preprint arXiv:2212.08073.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. Advances in neural information processing systems, 30.

De Andrade, I. M. and Tumelero, C. (2022). Increasing customer service efficiency through artificial intelligence chatbot. Revista de Gestão, 29(3):238–251.

Deo, A. and Murthy, K. (2023). Achieving efficiency in black-box simulation of distribution tails with self-structuring importance samplers. Operations Research.

Dong, J., Feng, M. B., and Nelson, B. L. (2018). Unbiased metamodeling via likelihood ratios. In 2018 Winter Simulation Conference (WSC), pages 1778–1789. IEEE.

Fan, W., Hong, L. J., and Zhang, X. (2020). Distributionally robust selection of the best. Management Science, 66(1):190–208.

Feng, B. M. and Song, E. (2019). Efficient input uncertainty quantification via green simulation using sample path likelihood ratios. In 2019 Winter Simulation Conference (WSC), pages 3693–3704. IEEE.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). Deep learning. MIT press.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction, volume 2. Springer.

He, S., Jiang, G., Lam, H., and Fu, M. C. (2023). Adaptive importance sampling for efficient stochastic root finding and quantile estimation. Operations Research.

Hong, L. J., Jiang, G., and Zhong, Y. (2022). Solving large-scale fixed-budget ranking and selection problems. INFORMS Journal on Computing, 34(6):2930–2949.

Hong, L. J. and Zhang, X. (2021). Surrogate-based simulation optimization. In Tutorials in Operations Research: Emerging Optimization Methods and Modeling Techniques with Applications, pages 287–311. INFORMS.

Hunter, D. R. (2004). Mm algorithms for generalized bradley-terry models. The annals of statistics, 32(1):384–406.

Johnson, R. and Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. Advances in neural information processing systems, 26.

Kim, S. and Henderson, S. G. (2007). Adaptive control variates for finite-horizon simulation. Mathematics of Operations Research, 32(3):508–527.

L. Salemi, P., Song, E., Nelson, B. L., and Staum, J. (2019). Gaussian Markov random fields for discrete optimization via simulation: Framework and algorithms. Operations Research, 67(1):250–266.

Li, D. and Sun, X. (2006). Nonlinear integer programming, volume 84. Springer.

Liu, G. (2015). Simulating risk contributions of credit portfolios. Operations Research, 63(1):104–121.

Luo, J., Hong, L. J., Nelson, B. L., and Wu, Y. (2015). Fully sequential procedures for large-scale ranking-and-selection problems in parallel computing environments. Operations Research, 63(5):1177–1194.

Nie, A., Zhang, Y., Amdekar, A. S., Piech, C., Hashimoto, T. B., and Gerstenberg, T. (2024). Moca: Measuring human-language model alignment on causal and moral judgment tasks. Advances in Neural Information Processing Systems, 36.

OpenAI (2023). Chatgpt. https://openai.com/blog/chatgpt.

Peherstorfer, B., Willcox, K., and Gunzburger, M. (2016). Optimal model management for multifidelity monte carlo estimation. SIAM Journal on Scientific Computing, 38(5):A3163–A3194.

Peng, Y., Xiao, L., Heidergott, B., Hong, L. J., and Lam, H. (2022). A new likelihood ratio method for training artificial neural networks. INFORMS Journal on Computing, 34(1):638–655.

Radiya-Dixit, E. and Wang, X. (2020). How fine can fine-tuning be? learning efficient language models. In International Conference on Artificial Intelligence and Statistics, pages 2435–2443. PMLR.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. (2024). Direct preference optimization: Your language model is secretly a reward model. Advances in Neural Information Processing Systems, 36.

Rhee, C.-h. and Glynn, P. W. (2015). Unbiased estimation with square root convergence for sde models. Operations Research, 63(5):1026–1043.

Semelhago, M., Nelson, B. L., Song, E., and Wächter, A. (2021). Rapid discrete optimization via simulation with Gaussian Markov random fields. INFORMS Journal on Computing, 33(3):915–930.

Shapiro, A., Dentcheva, D., and Ruszczynski, A. (2021). Lectures on stochastic programming: modeling and theory. SIAM.

Skiles, M. (2023). Ai for nonprofits: How to use artificial intelligence for good. https://donorbox.org/nonprofit-blog/ai-for-nonprofits. Accessed: 2023-12-27.

Tong, S. and Liu, G. (2016). Importance sampling for option greeks with discontinuous payoffs. INFORMS Journal on Computing, 28(2):223–235.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288.

Vihola, M. (2018). Unbiased estimators and multilevel monte carlo. Operations Research, 66(2):448–462.

Wang, X., Hong, L. J., Jiang, Z., and Shen, H. (2023). Gaussian process-based random search for continuous optimization via simulation. Operations Research.

Xie, W., Yi, Y., and Zheng, H. (2020). Global-local metamodel-assisted stochastic programming via simulation. ACM Transactions on Modeling and Computer Simulation (TOMACS), 31(1):1–34.

Xu, J., Zhang, S., Huang, E., Chen, C.-H., Lee, L. H., and Celik, N. (2014). Efficient multi-fidelity simulation optimization. In Proceedings of the Winter Simulation Conference 2014, pages 3940–3951. IEEE.

Xu, J. and Zheng, Z. (2023). Gradient-based simulation optimization algorithms via multi-resolution system approximations. INFORMS Journal on Computing, 35(3):633–651.

Zhang, P., Zeng, G., Wang, T., and Lu, W. (2024). Tinyllama: An open-source small language model.

Zheng, Z., Blanchet, J., and Glynn, P. W. (2018). Rates of convergence and clts for subcanonical debiased mlmc. In Monte Carlo and Quasi-Monte Carlo Methods: MCQMC 2016, Stanford, CA, August 14-19 12, pages 465–479. Springer.

Zheng, Z. and Glynn, P. W. (2017). A clt for infinitely stratified estimators, with applications to debiased mlmc. ESAIM: Proceedings and Surveys, 59:104–114.

Zhu, Y. and Dong, J. (2021). On constructing confidence region for model parameters in stochastic gradient descent via batch means. In 2021 Winter Simulation Conference (WSC), pages 1–12. IEEE.